

**Remarks/Arguments**

Claims 1-15 remain in this application.

The examiner deemed the drawings informal.

The examiner has rejected claims 1-15 under 35 USC 101 for not being concrete and tangible.

The examiner has rejected claims 1, 3 and 9-10 under 35 USC 102(b) as being anticipated by admitted prior art of record.

The examiner has rejected claims 2, 4-8 and 11-15 under 35 USC 103(a) as being unpatentable over admitted prior art of the programming languages of C and C++ for bit wise operations in view of the IDE of Visual C++ as documented by Ivor Horton, *Beginning Visual C++ 5*.

In view of the above amendments and these remarks, reconsideration of the above noted rejections and objections is respectfully requested.

**Drawing Objections:**

New formal drawings are attached hereto. Applicant respectfully submits that the new drawings cure the grounds for the objection.

**Rejections under 35 USC 101:**

The above amendments to independent claims 1 and 9 incorporate the examiner's suggested amendments. Applicant respectfully submits that these amendments overcome the rejection of claims 1-15 under 35 USC 101, as stated by the examiner.

**Rejections under 35 USC 102(b):**

Applicant respectfully traverses the rejection of claims 1, 3 and 9-10 under 35 USC 102(b). The independent claims are claims 1 and 9. Independent **claim 1** calls for:

for each bit-field, determining a first bit number defining a

position of the first bit of the bit-field within the variable;  
for each bit-field, determining a second bit number defining a  
position of the second bit of the bit-field within the variable; and  
for each bit-field, concatenating the first bit number and second  
bit number together to form a bit-field definition number defining a  
location of the bit-field within the variable.

Independent claim 9 calls for:

forming a bit-field definition number having a first portion and a  
second portion, the first portion formed from a location of the first bit in  
the machine word, the second portion formed from a location of the  
second bit in the machine word, the bit-field definition number defining  
a location and range of the bit-field in the machine word; and  
associating the bit-field definition number with the bit-field in the  
machine word.

Applicant respectfully submits that the prior art of record does not teach or suggest these limitations that describe how to form "bit-field definition numbers."

The prior art of record in the background section of the specification shows the normal way of handling bit-fields in the C and C++ programming languages. Specifically, a "structure" is "defined" to have N elements, referred to as "bit-fields." Only the length, or number of bits, of each bit-field is specified in the definition of the structure, as shown in Fig. 1 and on page 1 at lines 22-26. The location or position of each bit-field within the structure, however, is not specified. This lack of specifying bit-field location combined with the fact that different compilers assign bit-fields in different orders leads to the problem with bit order illustrated by Fig. 3 and described on page 2 at lines 6-16. In other words, the prior art of record does not describe how to form a "bit-field definition number" that defines the location of the bit-field within a variable, as called for in claim 1. Similarly, the prior art of record does not describe how to form a "bit-field definition number" that defines the location and range of the bit-field in a machine word, as called for in claim 9. Therefore, the prior art of record also does not teach or suggest how to determine first and second bit numbers that define positions of first and second bits of the bit-field within the variable, and which, when concatenated together, form the bit-field definition number and define the location of the bit-field within the variable, as called

for in claim 1. Likewise, the prior art of record does not teach or suggest first and second portions of a bit-field definition number formed from locations of first and second bits in a machine word, as called for in claim 9.

Applicant respectfully submits, therefore, that independent claims 1 and 9 are not anticipated by, are not obvious from, and are patentable over the prior art of record, since the prior art of record does not teach or fairly suggest the above-discussed claimed limitations. Additionally, since claims 3 and 10 depend from independent claims 1 and 9, respectively, claims 3 and 10 also are not anticipated by, are not obvious from, and are patentable over the prior art of record for the same reasons.

Rejections under 35 USC 103(a):

Applicant respectfully traverses the rejection of claims 2, 4-8 and 11-15 under 35 USC 103(a) as being unpatentable over admitted prior art of the programming languages of C and C++ for bit wise operations in view of the IDE of Visual C++ as documented by Ivor Horton, *Beginning Visual C++ 5*. Claims 2 and 4-8 depend from independent claim 1. Claims 11-15 depend from independent claim 9. As explained above, independent claims 1 and 9 are not anticipated by, are not obvious from, and are patentable over the admitted prior art. Furthermore, Applicant respectfully submits that *Beginning Visual C++ 5* does not overcome the deficiencies described above in the admitted prior art. In particular, *Beginning Visual C++ 5* does not describe how to form a "bit-field definition number" that defines the location of the bit-field within a variable, as called for in claim 1. Similarly, *Beginning Visual C++ 5* does not describe how to form a "bit-field definition number" that defines the location and range of the bit-field in a machine word, as called for in claim 9. Additionally, Applicant respectfully submits that claim 2 does not define "alias structures" or use "pointers to variables," but places the define directives for each bit-field within the associated variable's structure declaration, which makes the code look similar to the familiar bit-field declarations described in the background. Furthermore, Applicant respectfully submits that performing the operations of claims

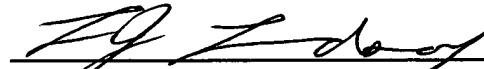
Appl. No. 10/021,606  
Amdt. Dated November 24, 2004  
Reply to Office action of September 23, 2004

4-8 and 11-15 with the claimed bit-field definition number is not taught or suggested in *Beginning Visual C++ 5*. Applicant respectfully submits, therefore, that dependent **claims 2, 4-8 and 11-15** are not anticipated by, are not obvious from, and are patentable over the prior art of record in view of *Beginning Visual C++ 5*.

For the reasons specifically discussed above, and others, it is believed that pending claims 1-15 define patentable subject matter. Reconsideration of the previous rejections as they might apply to the pending claims is therefore respectfully requested. Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

November 24, 2004  
Date

Respectfully submitted,



L. Jon Lindsay  
Registration No. 36,855  
ATTORNEY FOR APPLICANT

L. Jon Lindsay  
1857 W. Spring Water PL  
Highlands Ranch, CO 80129  
Telephone: 720-344-6189  
Facsimile: 928-563-4114